Access Grid 2.0

From Installation To Connection

Ti Leggett

Prerequisites

- Python 2.2
  - Win32 Extensions
  - Logging Module
- wxPython 2.4.0
- Globus
- http://www.mcs.anl.gov/fl/research/accessgrid/software/software.html

Python 2.2

Linux

- Ships with RedHat 7.3 and 8.0
- For earlier versions, build your own RPMs from SRPMs

Windows

- ActiveState Python comes standard with Win32 Extensions

wxPython

Linux

- Need wxGTK and wxPythonGTK

Windows

- Only need wxPython
- For development you'll want wxWindows as well

Globus

Linux

- gpt
- Globus
- pyGlobus

Windows

- WinGlobus
  - pyGlobus
  - pyDNS

Globus Linux Post Installation

- To finalize you must run `/usr/lib/globus/setup/setup-gsi` followed by `/usr/lib/globus/setup/globus_simple<blah>/setup-gsi –default`
- To start using you must either logout or source the proper globus shell script in `/etc/profile.d/globus.{sh|csh}`

Globus Certificates

Windows

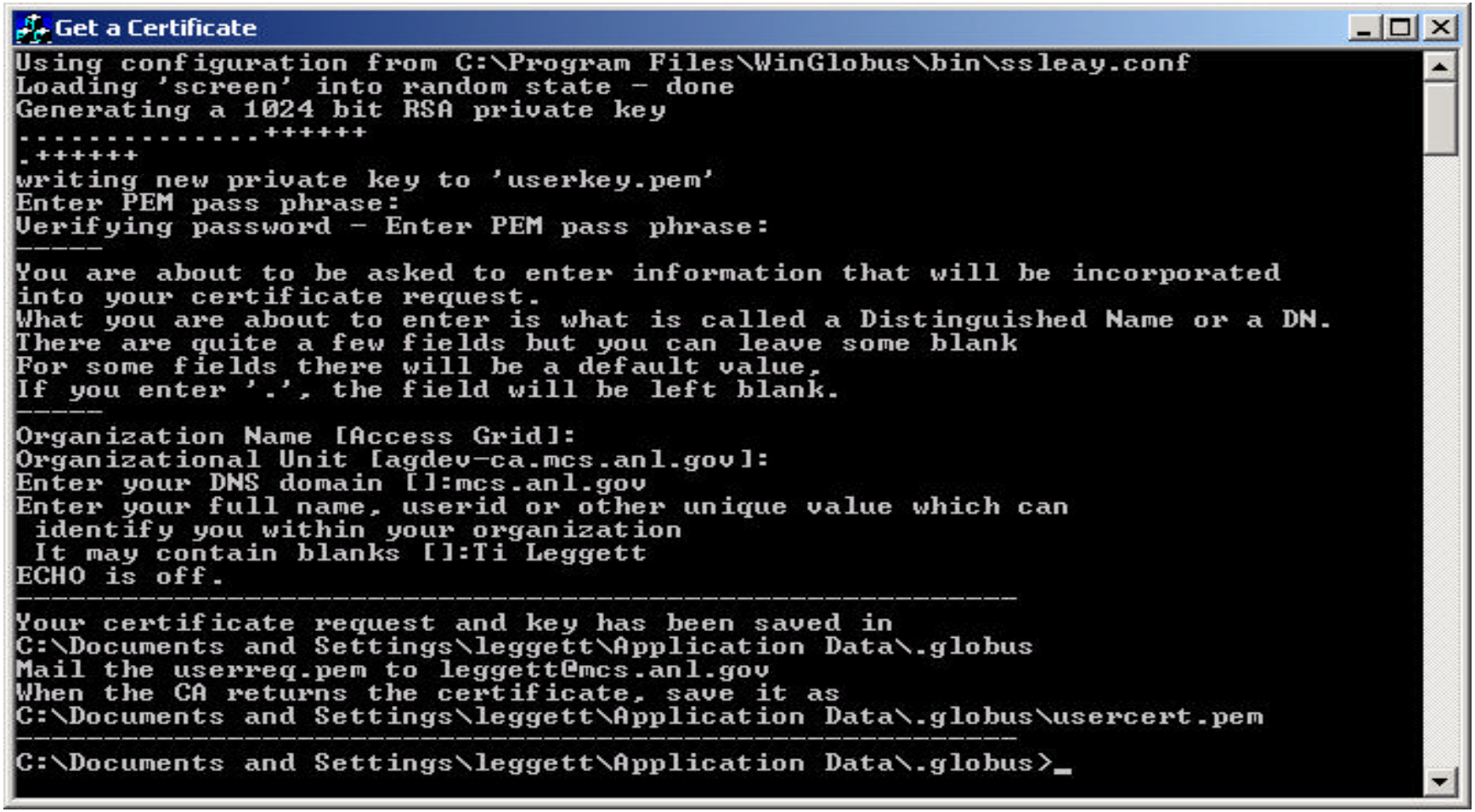
- Use the provided script to generate a request
  - %GLOBUS_LOCATION%\bin\certreq.cmd
  - Start\Programs\Windows Globus\Get a Certificate
  - Copy from a UNIX host using the Globus Configuration program

Globus Certificates

Windows (con’t)

- certreq.cmd
  - Use the defaults where provided
  - DNS domain (i.e., mcs.anl.gov)
  - Full Name (i.e., Ti Leggett)
  - Mail userreq.pem to the address reported back

Globus Certificates



Globus Certificates

Linux

```
$ grid-cert-request -force –cn
  “Ti Leggett”
```

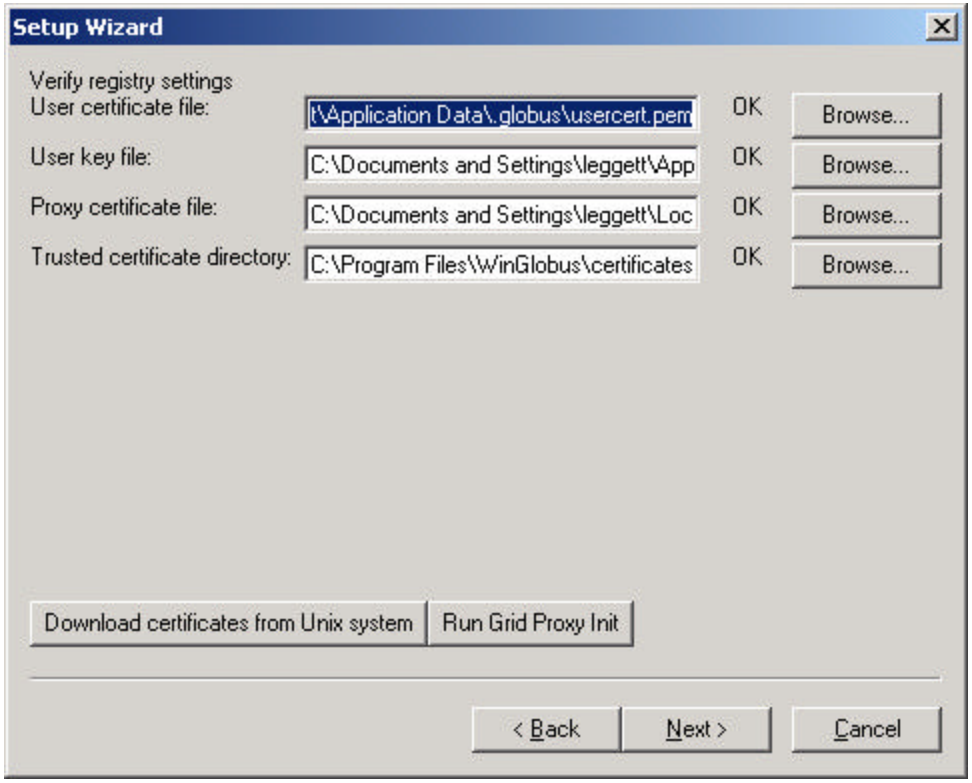
- Mail the ~/.globus/usercert_request.pem to the address reported back

Globus Certificates

Globus Certificates

```
/O=Access Grid/OU=agdev-
  ca.mcs.anl.gov/OU=mcs.anl.gov/C
  N=Ti Leggett
```

Setting up WinGlobus



Setting up WinGlobus

- User Certificate File
  - Where your personal certificate is kept
- User Key File
  - Where your private key is kept
- Proxy Certificate File
  - What file to store your proxy certificate
- Trusted Certficate Directory
  - Where your trusted certificates are kept
  - Comes with Globus (O=Globus) and Access Grid (O=Access Grid) certificates by default

Setting up Linux Globus

- Access Grid packaged Globus
  - /usr/lib/globus
  - Installs shell scripts to setup your Globus environment on login
  - Installs Access Grid (O=Access Grid) as default trusted certificate
- Pre-installed or User Installed Globus
  - You must set GLOBUS_LOCATION environment variable in your dot files
  - You must source in the proper Globus shell scripts for your shell
  - `bash$ . $GLOBUS_LOCATION/etc/globus-user-env.sh`

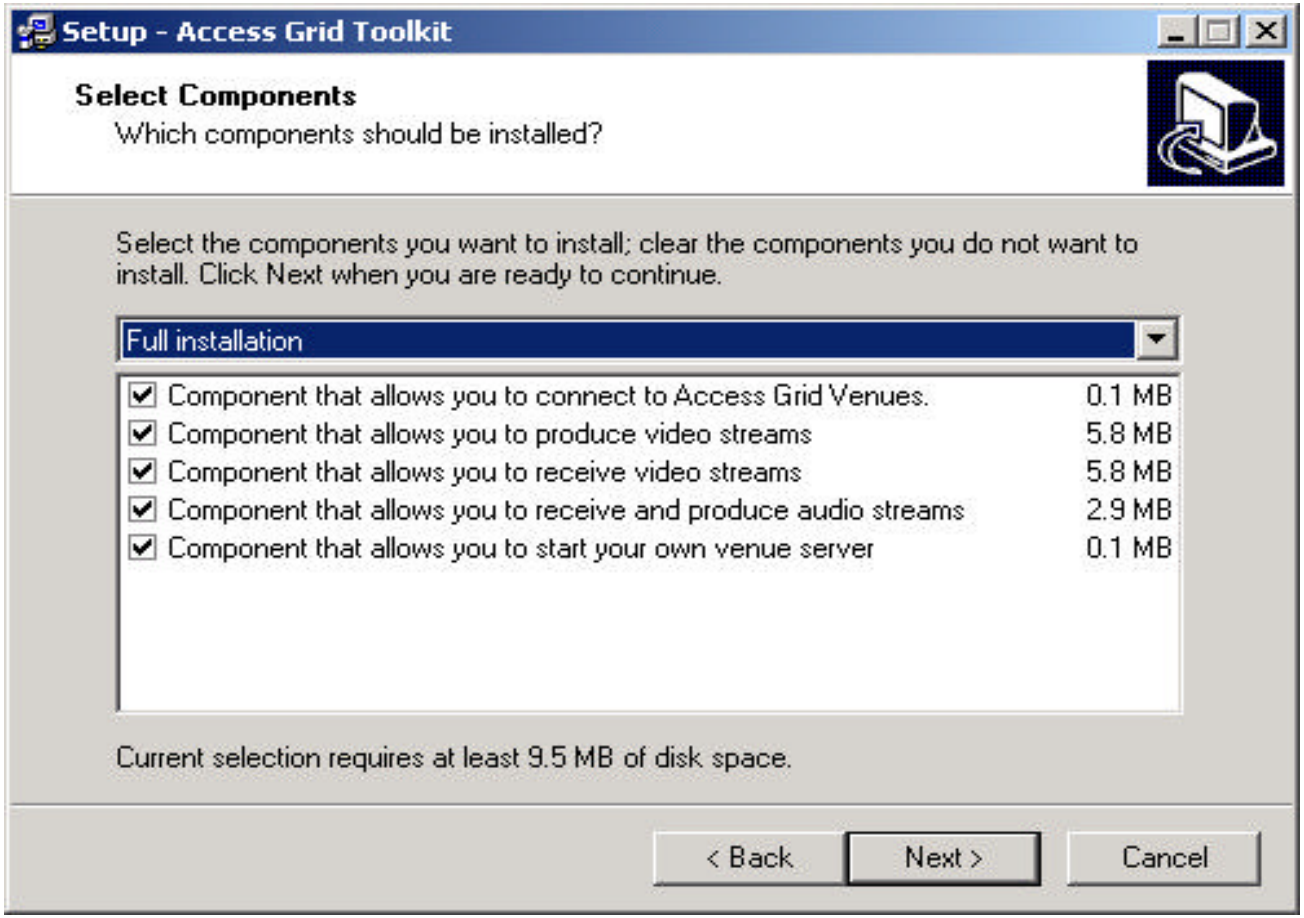
Access Grid Toolkit

- Core Modules
- Venue Client
- Venue Server
- Services
- Helper Applications
  - rat
  - vic

Core Modules

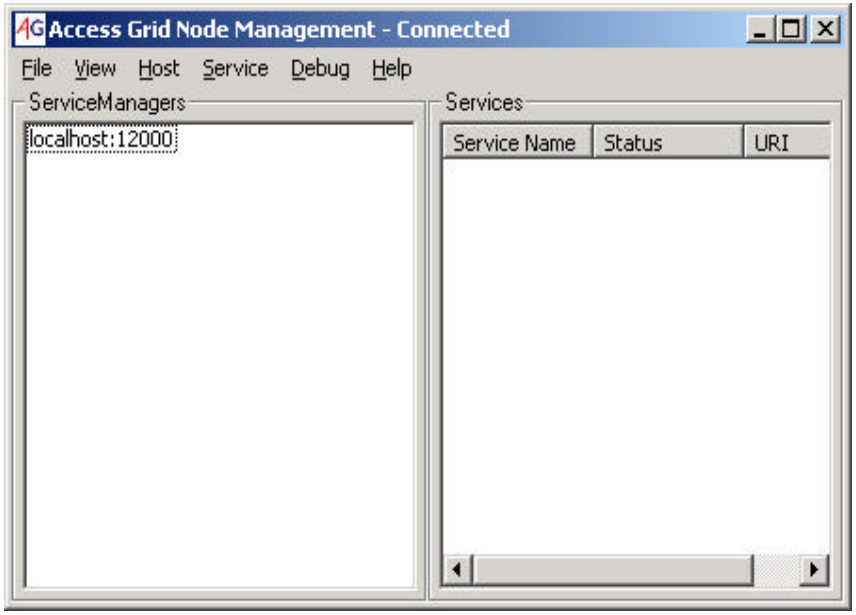
- Python Modules
- AG Service Manager
- Documentation

Installing


Setup - Access Grid Toolkit
Select Components
Which components should be installed?
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
Full installation
Component that allows you to connect to Access Grid Venues. 0.1 MB
Component that allows you to produce video streams 5.8 MB
Component that allows you to receive video streams 5.8 MB
Component that allows you to receive and produce audio streams 2.9 MB
Component that allows you to start your own venue server 0.1 MB
Current selection requires at least 9.5 MB of disk space.
< Back
Next >
Cancel

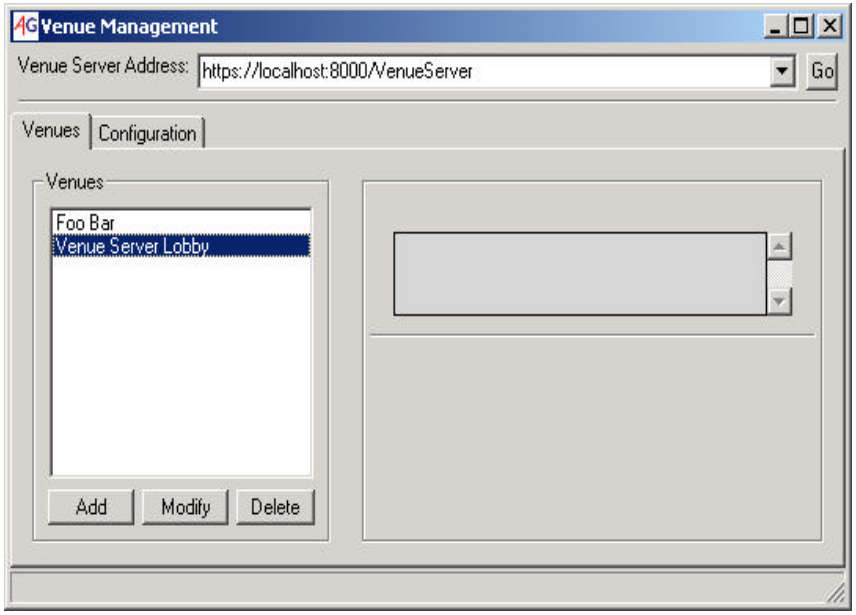
Venue Client

- AG Node Service
- Node Management
- Venue Client



Venue Server

- Venue Server
- Venue Management



Services

- Not necessary to install
- These will be pushed out to the Service Managers by the Node Manager as they are needed

Connecting with Access Grid 2.0

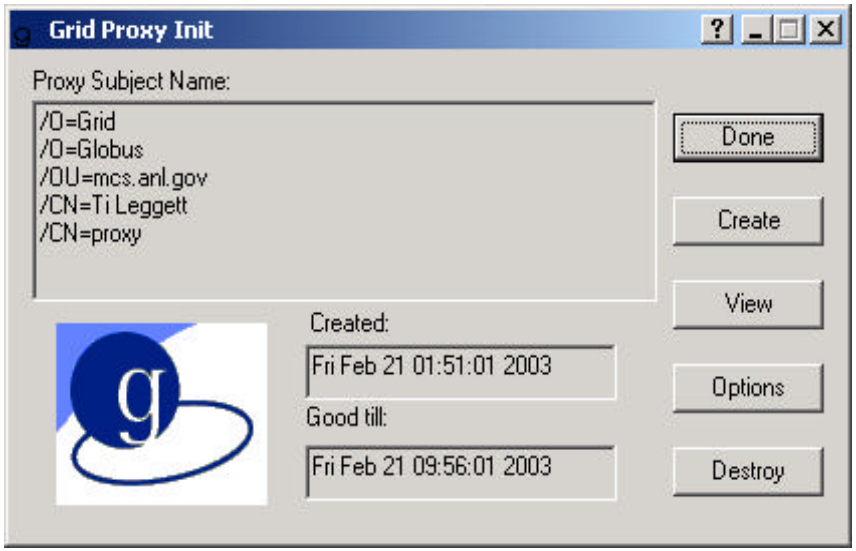
Getting a Proxy

Linux

- grid-proxy-init

Windows

- wgpi

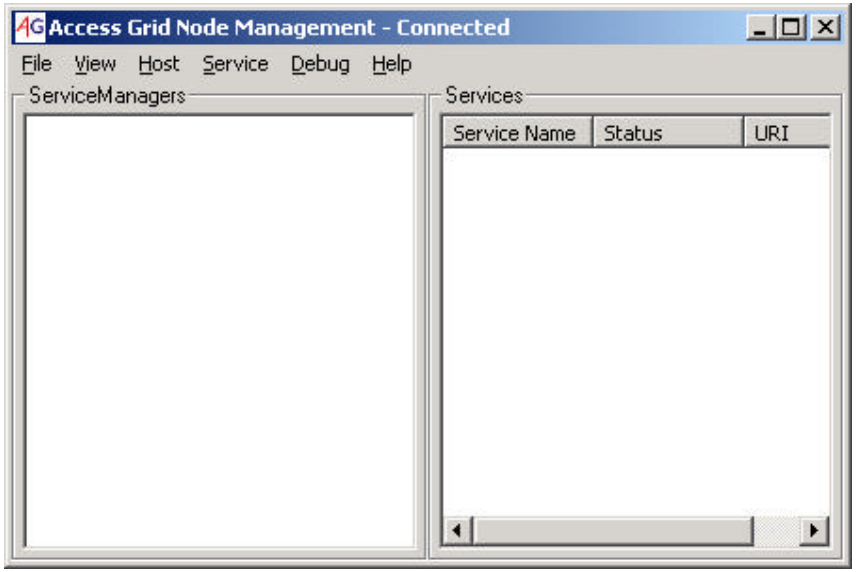


Starting Your Node

- AGNodeManager.py
  - start on your “control machine”
  - https://localhost:11000/NodeService
- AGServiceManager.py
  - start on each of your machines that will run services, including the Node Manager
  - https://<machine>:12000/ServiceManager
- Will be started as Windows services or linux daemons via /etc/rc.d/init.d
- Use Globus service certificates (CN=AGNodeService/ag.mcs.anl.gov)

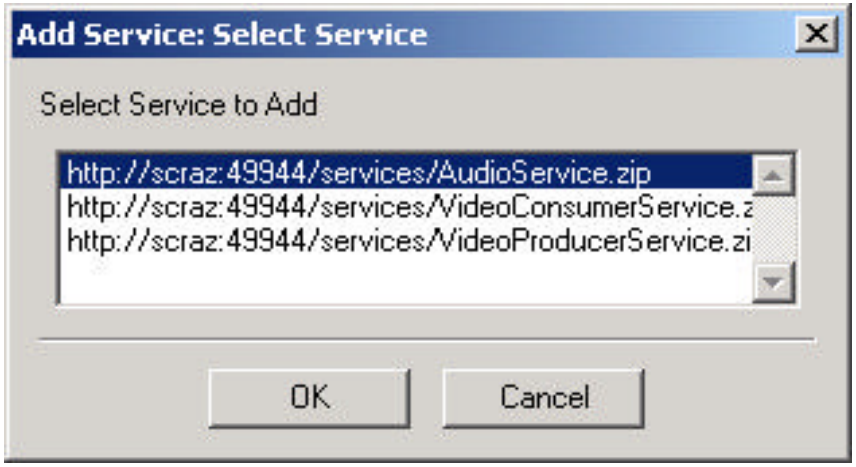
Node Management

- NodeManagement.py
- Uses user certificate for authentication
- Add Service hosts via `Host/Add Host` menu



Adding Services

- Use Node Manager
- Add Services to Service Managers by choosing the Manager and using the `Service/Add Service` menu
- Audio and Video Consumer services are not tied to a resource
- Use SetupVideo.py to configure Video Producer resources

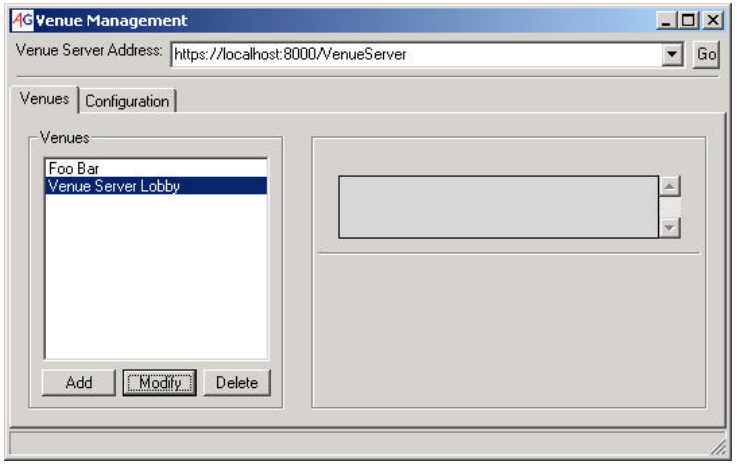


Starting a Venue Server

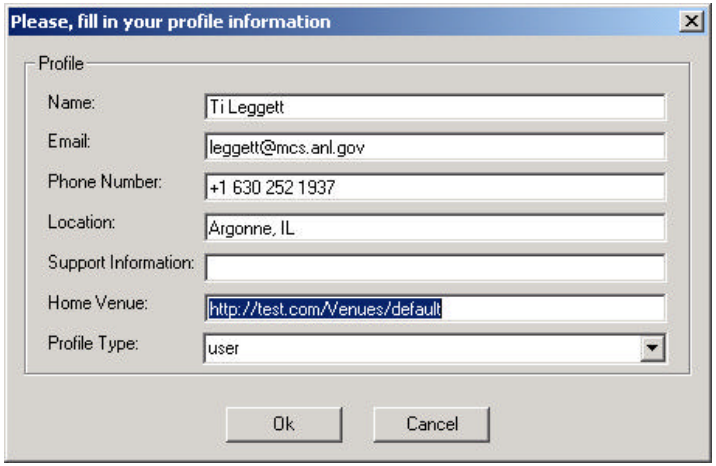
- VenueServer.py
  - https://venue.foo.com:8000/VenueServer
  - Started as Windows service or linux daemon via /etc/rc.d/init.d
  - Uses Globus service certificates

Managing Venues

- VenueManagement.py
  - Uses user certificate for authentication
  - Add venues, exits, descriptions, and multicast addressing schemes
  - Connect your Venues with remote Venues



Connecting to Venues



- VenueClient.py
- A transitional AG1.x Venue Server is at https://vv2.mcs.anl.gov:9000/Venues/default

Appendix

Linux Notes

- RPMs created on Red Hat 7.3
- Earlier versions of Red Hat do not have python 2.2 and must build python 2.2
- Red Hat 8.x might have problems because of library dependencies and Globus

Building RPMs from SRPMs

- Download the SRPM
- `rpm –-rebuild <name>.src.rpm`
- Writes RPMs to `/usr/src/redhat/RPMS/i386` or `/usr/src/redhat/RPMS/noarch`

Links

- Python
  - http://www.activestate.com/Products/ActivePython
  - <insert logging module link>
- wxWindows
  - http://www.wxwindows.org
- Globus
  - http://www.globus.org
- pyGlobus
  - http://www-itg.lbl.gov/gtg/projects/pyGlobus/
- pyDNS
  - http://pydns.sourceforge.net/